



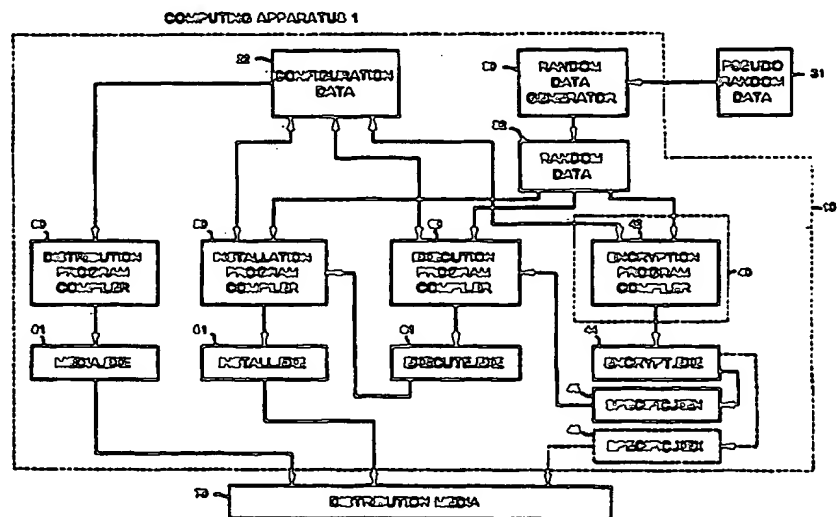
INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6: G06F 12/14, 19/00		A1	(11) International Publication Number: WO 96/18951
			(43) International Publication Date: 20 June 1996 (20.06.96)
(21) International Application Number: PCT/AU95/00836		(81) Designated States: AL, AM, AT, AU, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TT, UA, UG, US, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG), ARIPO patent (KE, LS, MW, SD, SZ, UG).	
(22) International Filing Date: 11 December 1995 (11.12.95)			
(30) Priority Data: PN 0025 13 December 1994 (13.12.94) AU			
(71)(72) Applicant and Inventor: DUNN, Alexander, Atkinson [GB/AU]; 17 Fairview Street, Hawthorn, VIC 3122 (AU).			
(74) Agent: CARTER SMITH & BEADLE; Qantas House, 2 Railway Parade, Camberwell, VIC 3124 (AU).		Published With international search report.	

(54) Title: METHODS AND APPARATUS FOR PROTECTION OF EXECUTABLE PROGRAMS, LIBRARIES AND DATA

(57) Abstract

A method of, and apparatus for, protecting a computer program from copying or propagation to other computer environments is provided in which an original executable program is encrypted by an encryption program compiler (42) into one or more encrypted program sections (45, 46), an execution program (61) for producing a decrypted image of the original executable program is compiled by an execution program compiler (60), an installation program (51) arranged to interact with the execution program (61) is compiled by an installation program compiler (50), and the arrangement is such that the execution program (61) includes at least one encrypted section (45) of the original executable program whereby the decrypted image of the original executable program can only be run in a target environment which has been installed with the execution program (61) and the installation program (51). The apparatus may also include a distribution program compiler (80) to compile a distribution program (81) for installing the installation program and execution program in the target computer environment. When the execution program is run in the target environment it rebuilds the original executable program in a controlled manner which helps to provide protection from viruses. The program compilers (40, 50, 60 and 80) may make use of random or pseudo-random data from a random data generator (30) and configuration data (22) with the installation, execution and distribution programs being tailored to particular target environments and/or to the source environment. Further features of the invention include the use of self-destructive programs and alias names for further security.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	GB	United Kingdom	MR	Mauritania
AU	Australia	GE	Georgia	MW	Malawi
BB	Barbados	GN	Guinea	NE	Niger
BE	Belgium	GR	Greece	NL	Netherlands
BF	Burkina Faso	HU	Hungary	NO	Norway
BG	Bulgaria	IE	Ireland	NZ	New Zealand
BJ	Benin	IT	Italy	PL	Poland
BR	Brazil	JP	Japan	PT	Portugal
BY	Belarus	KE	Kenya	RO	Romania
CA	Canada	KG	Kyrgyzstan	RU	Russian Federation
CF	Central African Republic	KP	Democratic People's Republic of Korea	SD	Sudan
CG	Congo	KR	Republic of Korea	SE	Sweden
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LU	Luxembourg	TD	Chad
CS	Czechoslovakia	LV	Larvia	TG	Togo
CZ	Czech Republic	MC	Monaco	TJ	Tajikistan
DE	Germany	MD	Republic of Moldova	TT	Trinidad and Tobago
DK	Denmark	MG	Madagascar	UA	Ukraine
ES	Spain	ML	Mali	US	United States of America
FI	Finland	MN	Mongolia	UZ	Uzbekistan
FR	France			VN	Viet Nam
GA	Gabon				

METHODS AND APPARATUS FOR PROTECTION OF EXECUTABLE PROGRAMS, LIBRARIES AND DATA

This invention relates to the protection of computer programs. It is particularly, but not exclusively, concerned with protecting executable programs, dynamic link libraries and data included in computer programs from unauthorised use or copying thereof.

Modern computer software is frequently supplied in a form which can readily be copied. The absence of means of protection has hitherto had a major effect on software development and its distribution. It is therefore desirable to provide a means of protecting software whereby the software may be executed in a particular target computer environment in such a manner that it cannot be propagated to further computer environments.

It is also desirable to provide a method of, and apparatus for, manufacturing computer programs which enables the programs to be distributed with an acceptable level of security.

It is further desirable to provide a system for protecting computer programs in which the propagation of viruses is substantially reduced.

According to a first aspect of the invention, there is provided a method of protecting a computer program from copying comprising the steps of:

encrypting an original executable program to produce an encrypted version of said original executable program;

compiling an execution program for producing a decrypted image of the original executable program from said encrypted version of the original executable program;

providing installation means for installing the execution program and said encrypted version of the original executable program into a target environment,

wherein the execution program includes at least one section of said encrypted version of the original executable program and the decrypted image of the original executable program can only be run in a target environment which has been installed with said execution program by said installation means.

For some applications, the execution program may include an entire

encrypted version of the original executable program, but more conveniently the execution program incorporates only an encrypted section of the original executable program, with remaining sections of the original executable program being distributed to a user. In the latter case, since at least one section of the original executable program is included in the execution program, an unauthorised person who only obtains access to the remaining sections is prevented from reconstructing the original executable program. The remaining program sections may be unencrypted, but preferably they are partially or wholly encrypted for greater security.

10 The installation means preferably includes an installation program which interacts with or incorporates part or all of the execution program whereby the installation program is arranged to create a modified execution program capable of reconstructing an image of the original executable file from the encrypted program section or sections. For further security, the installation program may be arranged
15 to be self-destructive or to be destroyed while it is run once to create the modified execution program.

The installation means may include a distribution program configured to install the installation program and execution program in a target computer environment.

20 The installation means, execution program and encrypted executable program may be distributed to users by any convenient means, for example either individually or collectively on data storage media such as disks, read-only memory, CD-roms, or by transmission media such as by satellite or radio-transmission or fibre optic cable.

25 The execution program, the installation program and/or the distribution program may include configuration data relating to the target environment in which the execution program is to be run and/or to the media used to distribute the programs to users. The present invention therefore provides a versatile system in which the distribution of executable programs to users can be controlled with the
30 installation means being tailored to the target environment in which the executable program is to be run and/or to the source environment for supplying the programs

to users.

According to a second aspect of the invention, there is provided apparatus for manufacturing encrypted software comprising encryption means to encrypt an original executable program to produce an encrypted version of the original executable program; execution program compilation means to compile an execution program for decrypting said encrypted version of the original executable program; installation program compilation means to compile an installation program for installing the execution program and said encrypted version of the original executable program in a target computer environment; wherein the installation program is arranged to interact with the execution program in such a manner that the execution program is not able to decrypt said encrypted version of the original executable program to produce a useful decrypted image of the original program unless the installation program has been run in the target computer environment.

Preferably, the apparatus for manufacturing the encrypted software comprises a computer which includes encryption compilation means to produce an encryption program for encrypting data from the original program to produce one or more encrypted program sections. At least one of said encrypted program sections may be input to the execution program compilation means to be included in the execution program.

In one form of the invention, the entire encrypted version of the original executable program may be input to the execution program compilation means to be included in the execution program. Alternatively, one or more of encrypted program sections may be included in the execution program with at least one further program section being stored in a file of program sections.

The encryption program compilation means preferably uses random or pseudo-random data produced by a random data generator in order to encrypt the program sections. As used herein, the term "encryption" encompasses within its scope encoding, expansion or compression such that subsequent decoding, compression or expansion is required to produce the executable decrypted image of the original program.

The encryption program compilation means may also use configuration data

from a configuration data file relating to the specific media source used to distribute the software, to the target computer environment in which the section or sections of the encrypted executable program are to be installed and/or relating to the particular application of the original executable program. The encryption program
5 compilation means therefore produces an executable encryption program which is specific to the application of the original program and/or its intended environment, and when the encryption program is run, it produces an output specific to the application. The installation program compiler and/or the execution program compiler may also make use of random data or pseudo-random data produced by
10 the random data generator and/or configuration data to produce the installation and execution programs respectively.

The encryption program compilation means is preferably adapted to update the configuration data when it produces said at least one encrypted program section. Similarly, the execution program compilation means may be adapted to update the
15 configuration data when it compiles the execution program. The execution program compiler and the installation program compiler can therefore make use of information created by the encryption program in order to create an execution program and an installation program respectively, each of which is unique to the particular application of the original program.

20 The output of the execution program compilation means is preferably used as input to the installation program compilation means so that the execution program or an encrypted version thereof may be incorporated within the installation program.

The apparatus preferably also includes distribution program compilation
25 means to compile a distribution program for installing the installation program and execution program in the target computer environment. The distribution program compilation means may make use of configuration data, preferably after it has been updated by the encryption, execution and installation programs, in order to create a distribution program which is unique to the particular application of the original
30 program.

The installation and execution programs and, when provided, the files of

encrypted program sections are made available for distribution to an end user for installation on the target computer, but the encryption program remains with the manufacturer and is not intended to be distributed to the user. The installation program and the distribution program may be distributed to the user separately from each other and from the file of encrypted program sections. Alternatively, the installation program, the execution program and, when provided, the file of encrypted program sections may be supplied to the user together, for instance on a common program storage means such as an installation disk, or by any convenient kind of transmission media.

According to a further important aspect of the invention, there is provided a self-destructive installation program, which is adapted to interact with an execution program to enable the execution program to read at least one encrypted program section of an original executable program to produce a decrypted image of the original program for utilization in a target computer environment, wherein the installation program is arranged to destroy itself while it is run once. After the installation program has been run and destroyed itself, it can no longer be propagated elsewhere. Furthermore, the file of encrypted program sections and the execution program are protected from being copied to, and used in, other computer environments since the execution program requires the installation program to enable it to produce a useful decrypted image of the original program. Also, at least one section of the encrypted original executable program and any related routines upon which it depends for satisfactory operation may be arranged to be internally self-destructive or to be destroyed or modified by the execution program while it is run in the target environment.

In accordance with another desirable feature of the invention, there is provided an execution program for decrypting encrypted program sections of an original executable program wherein the execution program is arranged to execute the decrypted image of the original program under an alias name. The reconstructed original executable program under the alias name may be arranged to be destroyed by the execution program or may itself be self-destructive providing security against the decrypted image of the original program and its execution

program being copied and used in another computer environment.

When the execution program is run in the user environment it rebuilds the original executable program by decrypting and re-assembling its various component sections. In this manner viruses which are added to any component will be excluded from the reconstruction and non-genuine components will result in failure to execute.

When the execution program is arranged to process program sections of the original program, it may modify, save or temporarily destroy some or all of those sections, for subsequent reinstatement. This controlled execution of the decrypted image of the original program helps to provide protection from infections, such as viruses, which do not appear when the program sections are re-instated.

A preferred embodiment of the present invention, will now be described, by way of example only, with reference to the accompanying drawings, in which:

Figure 1 is a schematic block diagram of computing apparatus for manufacturing encrypted software in accordance with the invention;

Figure 2 is a schematic flow chart showing the apparatus and procedures for the installation and use of the encrypted software;

Figure 3 is a block diagram of a process for generating pseudo-random data which may be used in the apparatus of Figure 1;

Figure 4 is a block diagram of a data conversion process for converting binary data to text format which may be used in the apparatus of Figure 1;

Figures 4a and 4b are block diagrams showing similar data conversion routines which may be used in the apparatus of Figure 1;

Figure 5 is a block diagram of the process used to compile the programs in the apparatus of Figure 1;

Figure 6 is a block diagram showing how the encryption program is run to produce encrypted program sections;

Figure 7 is a block diagram showing the process used to run the installation program in a target computer;

Figure 8 is a block diagram showing the process used to run the execution program in the target computer;

Figure 9 is a block diagram similar to that of Figure 1 showing a modified embodiment of apparatus in accordance with the invention;

Figure 10 is a flow chart similar to that of Figure 2 showing the apparatus and procedure used for installing encrypted software produced by the apparatus of
5 Figure 9.

The apparatus for manufacturing encrypted software shown in Figure 1 comprises a manufacturing computer 10 which includes a random data generator 30 for generating random or pseudo-random data 32 from an original file 31 of random data, encryption means 40 for encrypting an original executable file into
10 at least one encrypted program section 45 (SPECIFIC.XEN) and, optionally, one or more further program sections 46 (SPECIFIC.XEX), an installation program compiler 50 for compiling an installation program 51 (INSTALL.EXE), an execution program compiler 60 for compiling an execution program 61 (EXECUTE.EXE), and a distribution program compiler 80 for compiling a
15 distribution program 81 (MEDIA.EXE). The further program section or program sections 46 (SPECIFIC.XEX) may be unencrypted, or they may be partially or wholly encrypted depending upon the level of security required. For the sake of convenience, the following description will refer to encrypted program sections 46 (SPECIFIC.XEX).

20 As shown more particularly in Figure 5, the encryption means 40 includes an assembly level encryption compiler 42 provided with a source text 41 of an encrypt program and which uses random or pseudo-random data 32 from the random data generator 30 and configuration data from a configuration data file 22 to compile an encryption program 44 (ENCRYPT.EXE).

25 The configuration data file 22 used by the encryption compiler 42 to generate the encryption program 44 includes information preferably prepared in advance and relating specifically to the original file of binary data to be protected, to the source media to be used for the distribution of the programs and to a target computer environment in which the programs 61 (EXECUTE.EXE), 51
30 (INSTALL.EXE) and, optionally, 46 (SPECIFIC.XEX) are intended to be installed. For instance, the configuration data may include the full path to the source program

or library to be protected, the source path, the target path, an alias format, a selection table of environment factors to be checked on the distribution source and target computer environment and a strategy table for the processing input files of various sizes. The strategy table can determine whether or not it is necessary to
5 generate the further encrypted program sections 46 (SPECIFIC.XEX).

As shown in Figure 6, the encryption program 44 is arranged to encrypt the original executable program 12 into the first encrypted program section 45 (SPECIFIC.XEN) and, when required, further partly or wholly encrypted program section or sections 46 (SPECIFIC.XEX) which may be stored in files 48. The
10 encryption program may operate directly upon the original executable program 12 to convert the encrypted program sections 46 (SPECIFIC.XEX) to binary format which may then be stored in files 48. Alternatively, the manufacturing computer 10 may include a data converter for converting the encrypted program sections 46 (SPECIFIC.XEX) to binary text format. The encryption program compiler 42 is
15 able to update the configuration data file 22 with, for example, check total or sample encrypted code values for the files it has encrypted.

The encryption compiler 42 can make use of random data 32, or pseudo-random data 34 converted to text format generated in advance by the random data generator 30. As shown in Figures 3 and 4, the random data generator 30 may
20 generate pseudo-random data 34 from a file of random data 31 and the pseudo-random data may be stored in one or more files 36 or 18, possibly after passing at least some of the data through a data filter 38, before it is input to the encryption compiler 42. A data converter 20 may be used to convert the files 36 of binary data to files 18 of random or pseudo-random data in text format.

Referring to Figure 4A and Figure 6, the first encrypted program section 45 (SPECIFIC.XEN) produced by the encryption program 44 may be processed by a data conversion program 20A (CONVSPEC.EXE) (similar to the data converter 20) to produce a binary image 18A (SPECIFIC.RTN) in text format, which may be stored in a file 47 before being used as input to the execution program compiler 60
30 (Fig. 5). If the strategy table in the configuration data file 22 determines that further encrypted program sections 46 will be required, they may also be processed

by a data converter (not shown) similar to that of Figure 4A to produce encrypted program sections of binary data which can be stored in files 48 for subsequent distribution to a user, for instance by distribution media 70, such as an installation disk or by transmission media.

5 Referring to Figure 5 of the drawings, the execution program compiler 60 may also comprise an assembly level compiler provided with a source text 62 for the execute program, and having as input at least the first encrypted binary program section 18A (SPECIFIC.RTN) in text format, configuration data from file 22 and random (or pseudo-random) data 32 in text format. The configuration data 22
10 provided as input to the execution program compiler 60 may include path and alias or "skeleton" names which can be used when the program is executed in the target environment. The execution program compiler 60 preferably provides that successful execution of the execution program 61 (EXECUTE.EXE) is dependent on strict compliance therewith. By the use of configuration data 22, the
15 manufacturing computer 10 is thus able to create an execution program 61 which is unique to the particular application for the original executable file making use of information created by the encryption program 44 (ENCRYPT.EXE). The execution program 61 is then used as input to the installation program compiler 50 after being converted into text format 18B (SPECIFIC.RTX) by a data conversion
20 program 20B (CONVESEQ.EXE).

Referring also to Figure 5, the installation program compiler 50 may comprise an assembly level compiler provided with a source text 52 for the install program, and having as inputs the converted execution program in text format 18B, configuration data from file 22, and random (or pseudo-random) data in text format
25 32. The configuration data 22 which is input to the installation program compiler 50 may include an environment factor selection table that can determine which properties of the target environment have to be checked for propagation protection. Alternatively, the table may indicate that external proprietary routines are to be executed and results returned.

30 The configuration data file 22 is also adapted to receive information from the encryption program compiler 42, the execution program compiler 60 and the

installation program compiler 50. Thus, the file 22 of configuration data can be updated by the encryption program compiler 42 with data about the encryption program when the encryption program has been compiled, the updated configuration data being used by the execution program compiler 60 to compile the execution
5 program 61. Similarly, the execution program compiler 60 can update the configuration data file 22 with data about the execution program 61 which can then be used by the installation program compiler 50 in compiling the installation program 51. Likewise, the configuration data file 22 can be updated by the installation program compiler 50 with information about the installation program
10 51 which can be used by the distribution program compiler 80 in compiling the distribution program 81 (MEDIA.EXE)

Referring to Figure 5 the distribution program compiler 80 may comprise an assembly level compiler provided with a source text 82 for the distribution program (MEDIA.EXE), and having as inputs configuration data from file 22, and random
15 (or pseudo-random) data 32. The distribution program, compiled last in the sequence of compilations with configuration data 22 as input is thereby provided with additional information useful to decide on alternative courses of action for distributing the software from the source to the target environment.

The data conversion programs 20A CONVSPEC.EXE and 20B
20 CONVEXEQ.EXE may involve an element of data conversion or encryption in addition to their function to produce binary data in text format suitable to be read by a computer compiler.

The source code for the compiler programs which make use of the configuration data may direct that only selected parts of the configuration data will
25 be embodied in the output compilation, and conversely may direct that selected parts of the configuration data may be updated as a result of the compilation.

The strategy table in the configuration data file 22 is somewhat similar to an object in computer terminology in that it contains both addresses of functions and data. The data which may be returned into the strategy table in the process of
30 compilation may be information such as computed check sums or parts of encryption keys to be passed on to subsequent compilations in the sequence.

Encryption keys can thus be incremental, originating say from media identification, program serial identification and feedback information introduced into the configuration data during earlier compilations. The distribution program 82 (MEDIA.EXE) being the last in the compilation sequence can be aware of and
5 make use of all that precedes it.

The encrypted software consisting of the installation program 51 (INSTALL.EXE), the distribution program 81 (MEDIA.EXE) and the file or files 46 of encrypted program sections (SPECIFIC.XEX) may be transferred to an installation disk 70 or other file storage means for supply to a distributor or user.
10 The encryption program (ENCRYPT.EXE) remains with the manufacturer and is not intended to be distributed to the user.

The encrypted software on the distribution media 70 can be installed and used in a target environment of an installing agent or user by following the procedures illustrated with reference to Figures 2, 7 and 8. The distribution
15 program 81 (MEDIA.EXE), is run to transfer the installation program 51 (INSTALL.EXE) and the file or files 48 (if present) of encrypted program sections (SPECIFIC.XEX) from the distribution media 70 to the target environment. The program 81 (MEDIA.EXE) may convert or revise the installation program 51 (INSTALL.EXE) to make it dependent on features of the target environment for
20 successful subsequent operation.

In accordance with one installation procedure, the program 81 (MEDIA.EXE) can read the distribution media 70, copy or transfer the relevant files to the target environment, run the install program 51 (INSTALL.EXE) leaving a modified execution program 71 (EXECUTE.EXE) in the target environment. The
25 end user can then run the modified execution program 71 (EXECUTE.EXE) to reconstruct an image 74 of the original executable program and run the application.

In another installation procedure, for additional protection the modified installation program 51 (INSTALL.EXE) may itself be encrypted by these disclosed methods or by external means. The resultant encrypted version 151 of the install
30 program may be copied to a master disk which may, for example be distributed to an installer. When the installer decrypts the installation routine to the target

environment the distribution program 81 (MEDIA.EXE) can sense that the correct version of the installation program is present in the environment and proceed with the installation.

In this manner, the manufacturer of the encrypted software can control the
5 distribution of the software, and propagation of the encrypted software is substantially reduced.

In accordance with another advantageous feature of the invention, further protection may also be provided by arranging one or more of the installation program 51 (INSTALL.EXE), the modified execution program 71 and the
10 distribution program 81 (MEDIA.EXE) to be self-destroying, run once programs, as illustrated in Figures 2 and 7. For example, while the installation program 51 (INSTALL.EXE) is run in the target environment it may be arranged to destroy itself while modifying the execution program 61 (EXECUTE.EXE) to produce the modified execution program 71. Subsequent copying of the installation program
15 51 (INSTALL.EXE), which is required to enable the execution program to decrypt and restore the original program sections and rebuild the original executable program, is therefore prevented.

Referring more specifically to Figure 8, the modified execution program 71 (EXECUTE.EXE) includes first decryption means 72 to, decrypt and restore a first
20 section of the original executable program internally within itself, second decryption means 73 to decrypt and restore other sections of the original program externally, and reconstruction means 74 to concatenate the decrypted sections and rebuild the image of the original executable program.

In accordance with a further advantageous feature of the invention, the
25 execution program (EXECUTE.EXE) includes alias assignment means 75 for loading and executing the restored image of the original executable under an alias name. The alias program may be arranged to be self destructive when run once, or the execution program (EXECUTE.EXE) may include means 78 arranged to destroy the alias program when run. The names and extensions given to files of all
30 kinds in these descriptions are for illustrative purposes only, the configuration file 22 determines the actual names which will be used for each particular application.

The ability to use such covert alias names provides further protection from targeted viruses. Executable programs such as those designated with a suffix .EXE are often supported by other routines which they depend upon for their operation. References to such executable programs should be taken to include such supporting routines and their data.

The execution program (EXECUTE.EXE) may also include means 76 for destroying program sections and input data, and reconstruction means 77 capable of rebuilding and reinstating destroyed sections. The execution program may have the ability to recognise a different course of action for dynamic link library files.

10 The execution program can support parameters when run in the target environment. These may be passed to the alias program which the execution program executes under its control.

The installation program (INSTALL.EXE) and the execution program (EXECUTE.EXE) are preferably constructed such that they run through to completion whether or not they produce useful output. They are preferably arranged such that no error messages, which may be helpful in revealing the programs are generated. The encryption program 44 (ENCRYPT.EXE) is preferably arranged to encrypt the program sections of the original executable program such that there are no vacant buffer areas or sequences of identical data in the unencrypted source files for INSTALL.EXE and EXECUTE.EXE, these being filled with random or pseudo-random data generated by the random data generator 10, 30. Encryption of the sections of programs may be overlapping, and to more than one level of depth.

Whilst no encryption system can be said to be completely secure from decryption and copying, the present invention provides a method of and apparatus for manufacturing encrypted software in which protection of an original executable program from copying is substantially increased and in which the encrypted software has increased protection from viruses and intruders. Furthermore whilst the protection system may appear complex, this occurs in the manufacturing process which can readily be automated and in practise the user will be unaware that the original application software is protected. Dependant on the level of protection

required, not all steps of the manufacturing sequence may be required during a production run.

It will be appreciated that various modifications and alterations to the system described above with reference to Figures 1 to 8 of the drawings may be made without departing from the scope or spirit of the invention. For instance, a common assembly level compiler in the manufacturing computer may be used to compile the encryption program (ENCRYPT.EXE), the installation program (INSTALL.EXE) and the execution program (EXECUTE.EXE). Also, instead of being incorporated wholly within the installation program 51, the execution program 61 may be transferred to the target environment separately from the installation program 51 as illustrated in the modified embodiment of Figures 9 and 10.

Figures 9 and 10 are similar to Figures 1 and 2 respectively and corresponding reference numerals have been applied to corresponding parts. Figure 10 differs from Figure 1 in that the execution file 61 is not used as input to the installation program compiler 50, and Figure 10 differs from Figure 2 in that when the installation program 51 and the execution program 61 are installed in the target environment the installation program 51 (INSTALL.EXE) is arranged to read the execution program 61 (EXECUTE.EXE) and interact with it to produce the modified execution program 71.

Claims:

1. A method of protecting a computer program from copying comprising the steps of:

encrypting an original executable program to produce an encrypted version
5 of said original executable program;

compiling an execution program for producing a decrypted image of the original executable program from said encrypted version of the original executable program;

providing installation means for installing the execution program and said
10 encrypted version of the original executable program into a target environment,

wherein the execution program includes at least one section of said encrypted version of the original executable program and the decrypted image of the original executable program can only be run in a target environment which has been installed with said execution program by said installation means.

15 2. A method according to claim 1 wherein the execution program includes an entire encrypted version of the original executable program.

3. A method according to claim 1 wherein the execution program includes only an encrypted section of the original executable program, and remaining sections of the original executable program are distributed to a user.

20 4. A method according to claim 3 wherein the remaining program sections are partially or wholly encrypted.

5. A method according to any one of the preceding claims wherein the installation means includes an installation program which interacts with or incorporates part or all of the execution program whereby the installation program
25 is arranged to create a modified execution program capable of reconstructing an image of the original executable file from the encrypted program section or sections.

6. A method according to claim 5 wherein the installation program is arranged to be self-destructive or to be destroyed while it is run once to create the
30 modified execution program.

7. A method according to any one of the preceding claims wherein at

least one section of the encrypted original executable program is arranged to be self-destructive or to be destroyed or modified by the execution program while it is run in the target computer environment.

8. A method according to any one of the preceding claims wherein the
5 installation means includes a distribution program configured to install the installation program and execution program in a target computer environment.

9. A method according to claim 8 wherein the execution program, the
installation program and/or the distribution program may include configuration data relating to the target environment in which the execution program is to be run
10 and/or relating to the source environment used to distribute the programs to users.

10. A method according to any one of the preceding claims further comprising the step of using random or pseudo-random data to encrypt the original executable program.

11. A method according to claim 5 or claim 6 wherein random or pseudo-
15 random data is used in the production of the installation program.

12. A method according to claim 8 or claim 9 wherein random or pseudo random data is used in the production of the distribution program.

13. A method according to any one of the preceding claims wherein the execution program is arranged to execute the decrypted image of the original
20 program under an alias name.

14. A method according to claim 13 wherein the reconstructed original executable program under the alias name is arranged to be destroyed by the execution program or is self-destructive providing security against the decrypted image of the original program and its execution program being copied and used in
25 another computer environment.

15. A method according to any one of the preceding claims wherein the execution program is arranged to rebuild the original executable program by decrypting and re-assembling encrypted program sections of the original executable program.

30 16. A method according to claim 15 wherein the execution program is arranged to modify, save or temporarily destroy at least one of said encrypted

program sections, for subsequent reinstatement, when processing the encrypted program sections.

17. Apparatus for manufacturing encrypted software comprising:
encryption means to encrypt an original executable program to produce an
5 encrypted version of the original executable program; execution program
compilation means to compile an execution program for decrypting said encrypted
version of the original executable program; installation program compilation means
to compile an installation program for installing the execution program and said
10 encrypted version of the original executable program in a target computer
environment; wherein the installation program is arranged to interact with the
execution program in such a manner that the execution program is not able to
decrypt said encrypted version of the original executable program to produce a
useful decrypted image of the original program unless the installation program has
been run in the target computer environment.

15 18. Apparatus according to claim 17 comprising a computer including
encryption compilation means to produce an encryption program for encrypting data
from the original program to produce a plurality of encrypted program sections.

19. Apparatus according to claim 18 wherein at least one of said
20 encrypted program sections is input to the execution program compilation means
to be included in the execution program.

20. Apparatus according to any one of claims 17 to 19 wherein the entire
encrypted version of the original executable program is input to the execution
program compilation means for inclusion in the execution program.

21. Apparatus according to claim 17 wherein at least one encrypted
25 program section is stored in a file of program sections instead of being input to the
execution program compilation means.

22. Apparatus according to any one of claims 17 to 21 further comprising
a random data generator for generating random or pseudo-random data.

23. Apparatus according to claim 22 wherein the encryption program
30 compilation means uses random or pseudo-random data produced by said random
data generator to encrypt the original executable program.

24. Apparatus according to claim 22 or claim 23 wherein the installation program compilation means uses random data or pseudo-random data produced by said random data generator when producing the installation program.

25. Apparatus according to any one of claims 22 to 24 wherein the
5 execution program compiler uses random or pseudo-random data produced by said random data generator when producing the execution program.

26. Apparatus according to any one of claims 17 to 25 further comprising data storage means including a configuration data file relating to one or more of the following: the specific media source used to distribute the software; the target
10 computer environment in which the section or sections of the encrypted executable program are to be installed; and/or the particular application of the original executable program.

27. Apparatus according to claim 26 wherein the encryption program compilation means uses configuration data from said configuration data file when
15 encrypting said original executable program.

28. Apparatus according to claim 26 or claim 27 wherein the installation program compilation means uses configuration data from said configuration data file when producing the installation program.

29. Apparatus according to any one of claims 26 to 28 wherein the
20 execution program compilation means uses configuration data from said configuration data file when producing the execution program.

30. Apparatus according to claim 27 wherein the encryption program compilation means is adapted to update the configuration data when it produces said at least one encrypted program section.

25 31. Apparatus according to claim 29 wherein the executed program compilation means is adapted to update the configuration data when it compiles the execution program.

32. Apparatus according to any one of claims 17 to 31 wherein the output of the execution program compilation means is used as input to the installation
30 program compilation means so that the execution program or an encrypted version thereof can be incorporated within the installation program.

33. Apparatus according to any one of claims 17 to 32 further comprising distribution program compilation means to compile a distribution program for installing the installation program and execution program in the target computer environment.

5 34. Apparatus according to claim 33 as appended to any one of claims 26 to 31 wherein the distribution program compilation means uses configuration data from said configuration data file in order to create a distribution file which is unique to the particular application of the original executable program.

10 35. A self-destructive installation program adapted to interact with an execution program to enable the execution program to read at least one encrypted program section of an original executable program to produce a decrypted image of the original program for utilisation in a target computer environment, wherein the installation program is arranged to destroy itself while it is run once.

15 36. An execution program for decrypting encrypted program sections of an original executable program wherein the execution program is arranged to execute the decrypted image of the original executable program under an alias name.

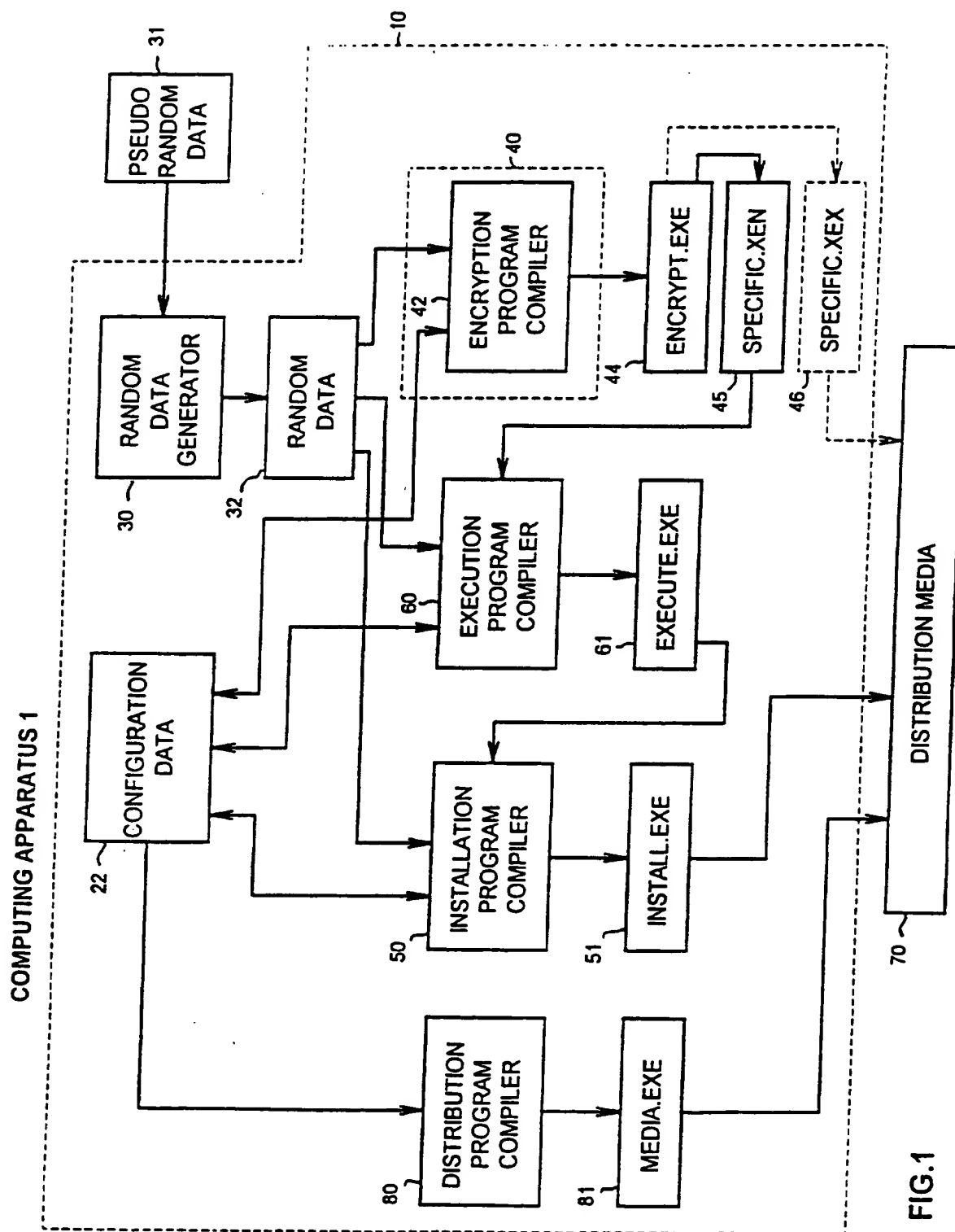


FIG.1

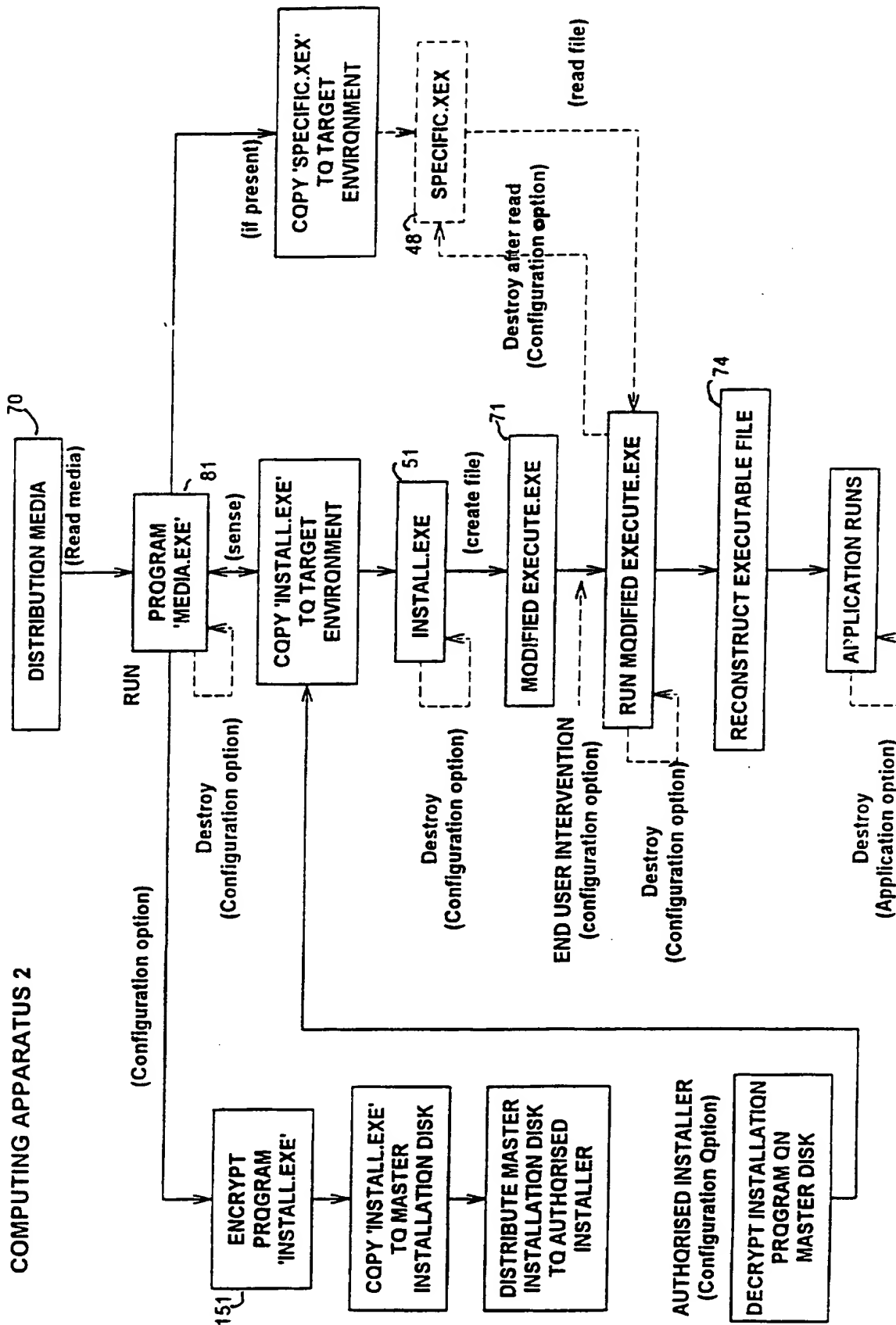
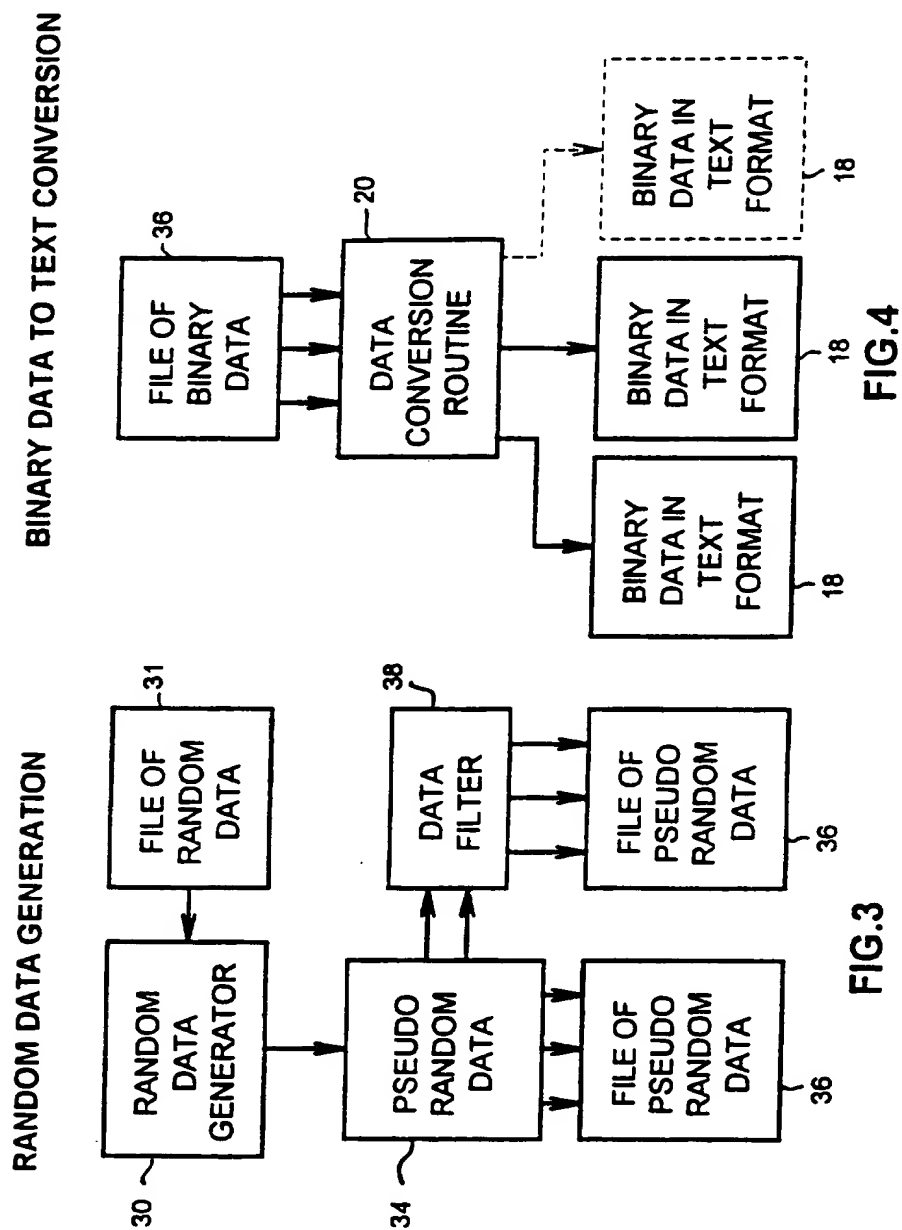


FIG. 2



4 / 1 0

BINARY DATA TO TEXT CONVERSION

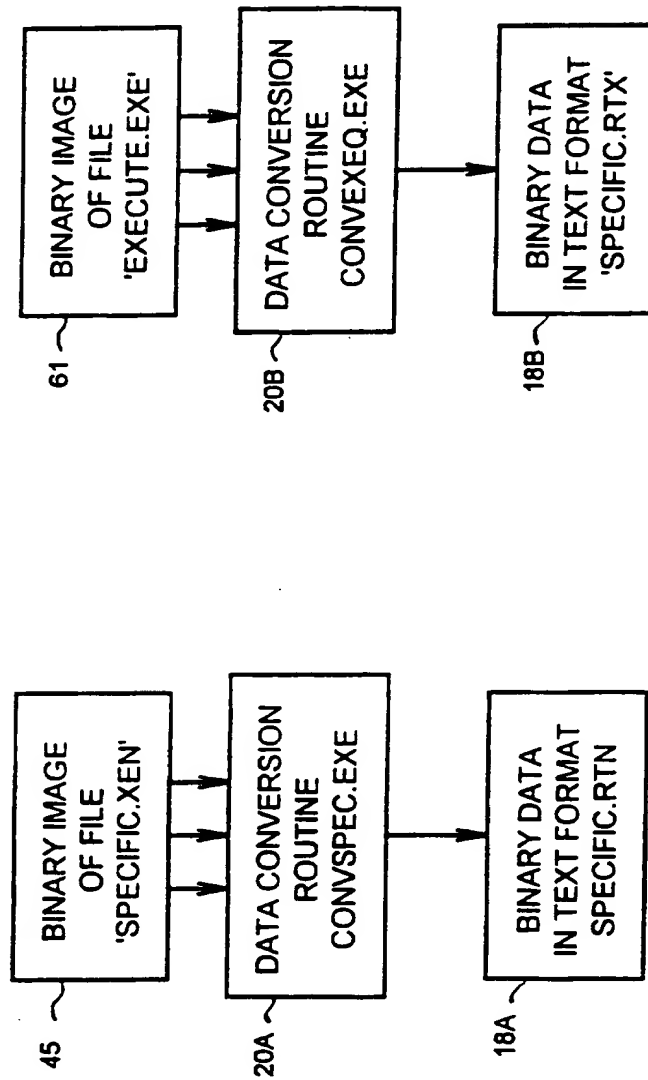
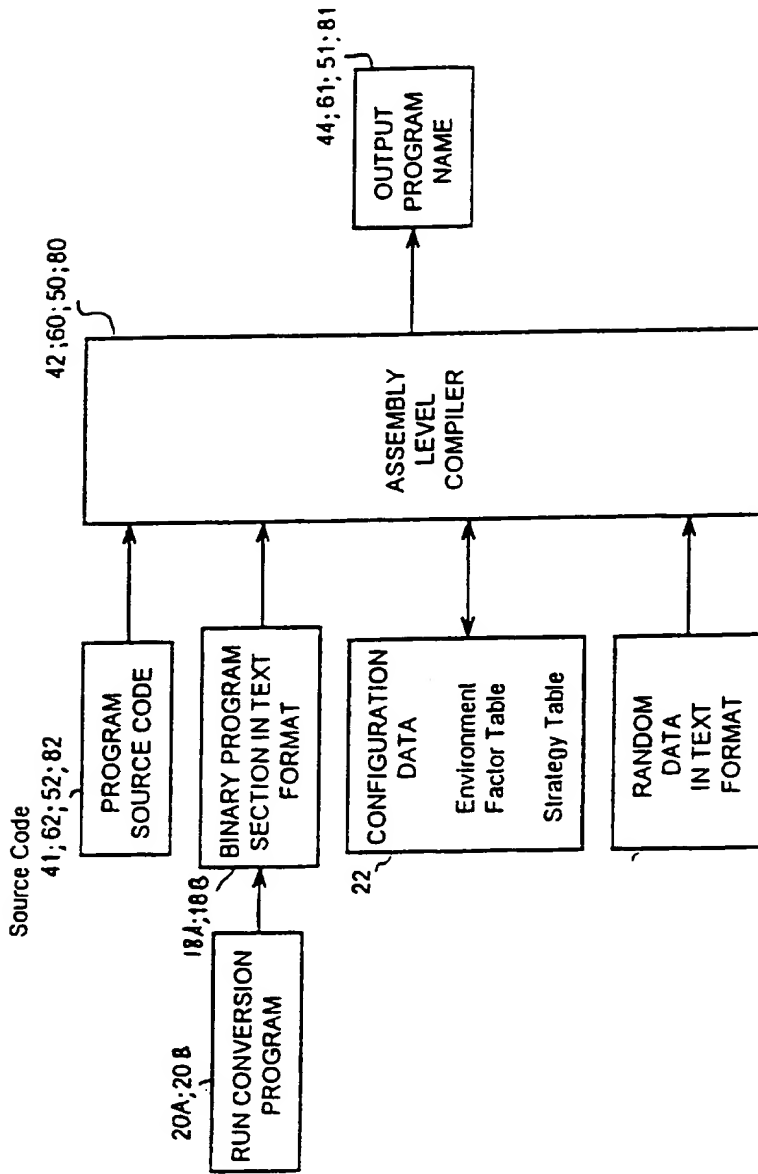


FIG. 4A

FIG. 4B

ROUTINES TO COMPILE PROGRAMS



Conversion Program Name	Program Source Code	Output Program Name	Compiler Reference
Not Applicable	Ref 41 "ENCRYPT.ASM"	Ref 44 "ENCRYPT.EXE"	Ref. 42
Ref 20 "CONVSPEC.EXE"	Ref 62 "EXECUTE.ASM"	Ref 61 "EXECUTE.EXE"	Ref. 60
Ref 20 "CONVESEQ.EXE"	Ref 52 "INSTALL.ASM"	Ref 51 "INSTALL.EXE"	Ref. 50
Not Applicable	Ref 82 "MEDIA.ASM"	Ref 81 "MEDIA.EXE"	Ref. 80

FIG.5

6 / 1 0

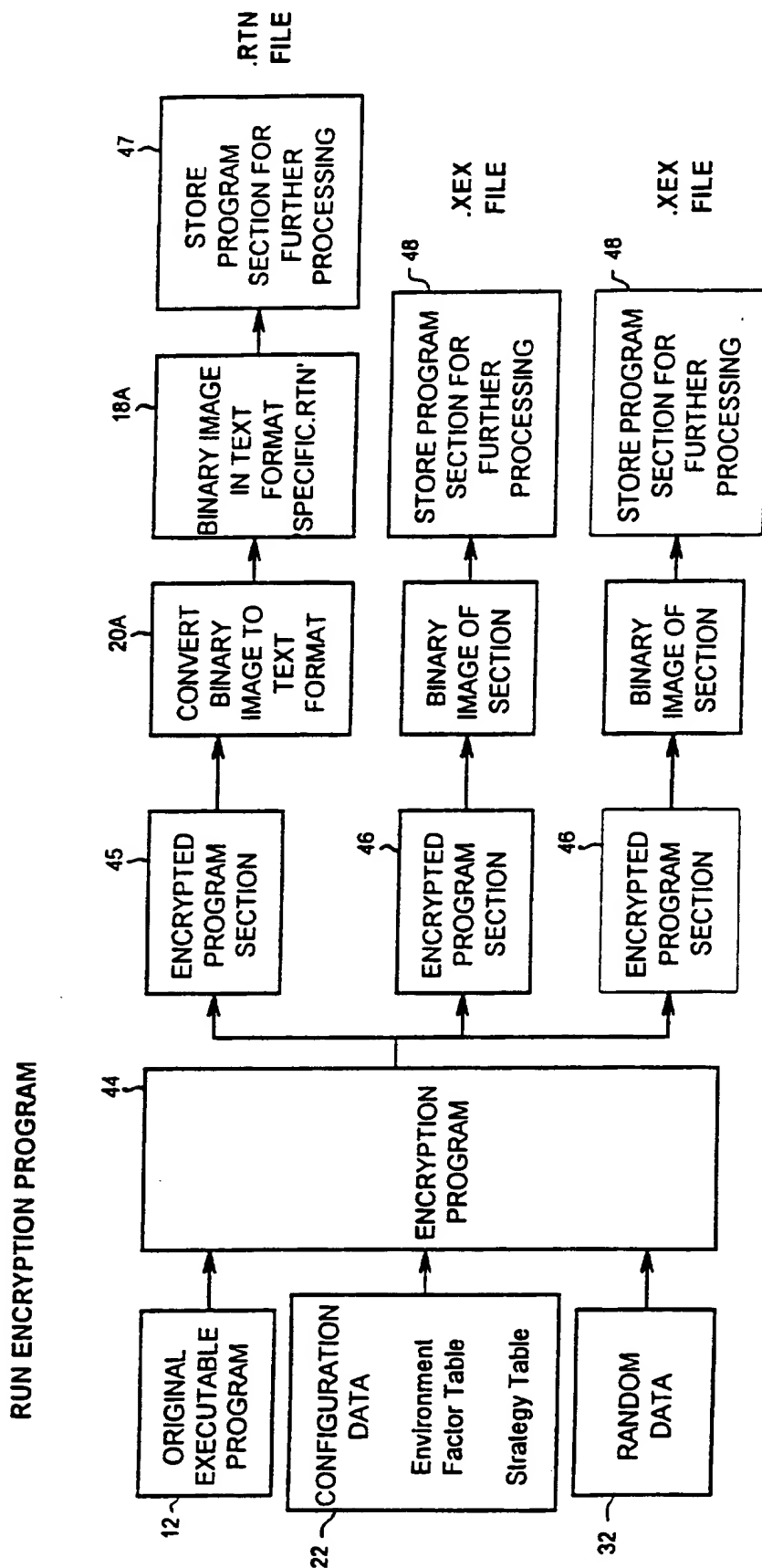


FIG.6

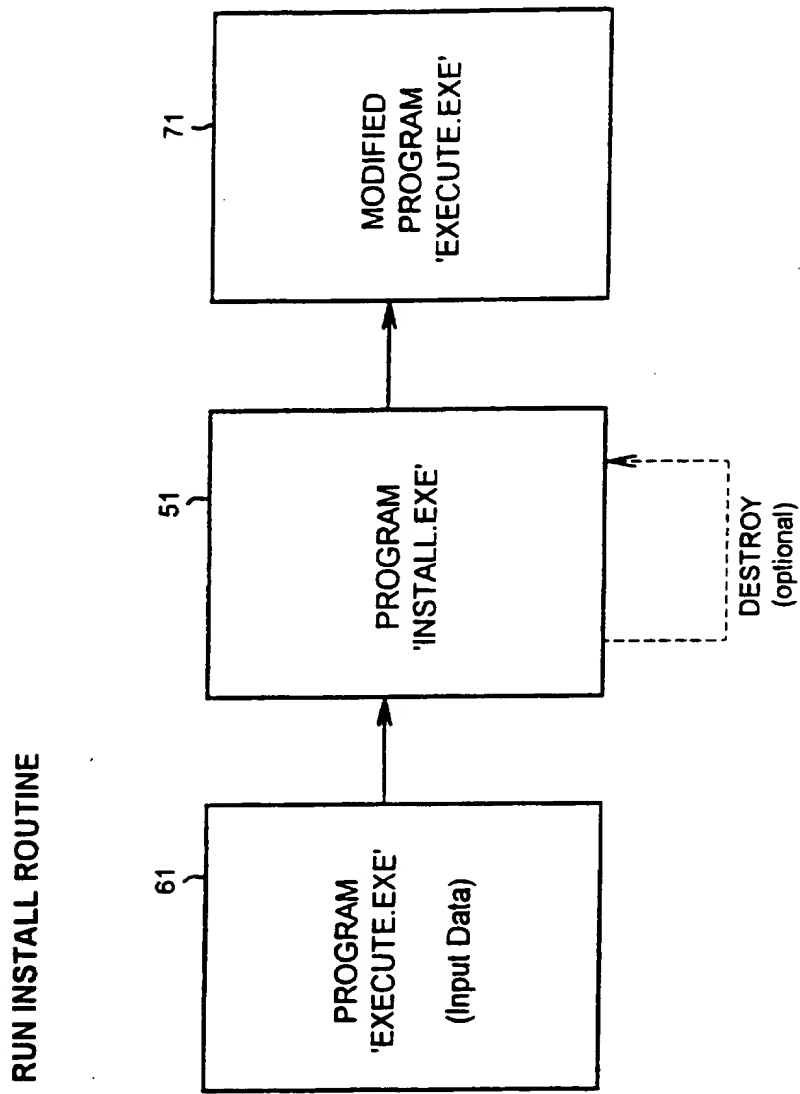


FIG. 7

8 / 10

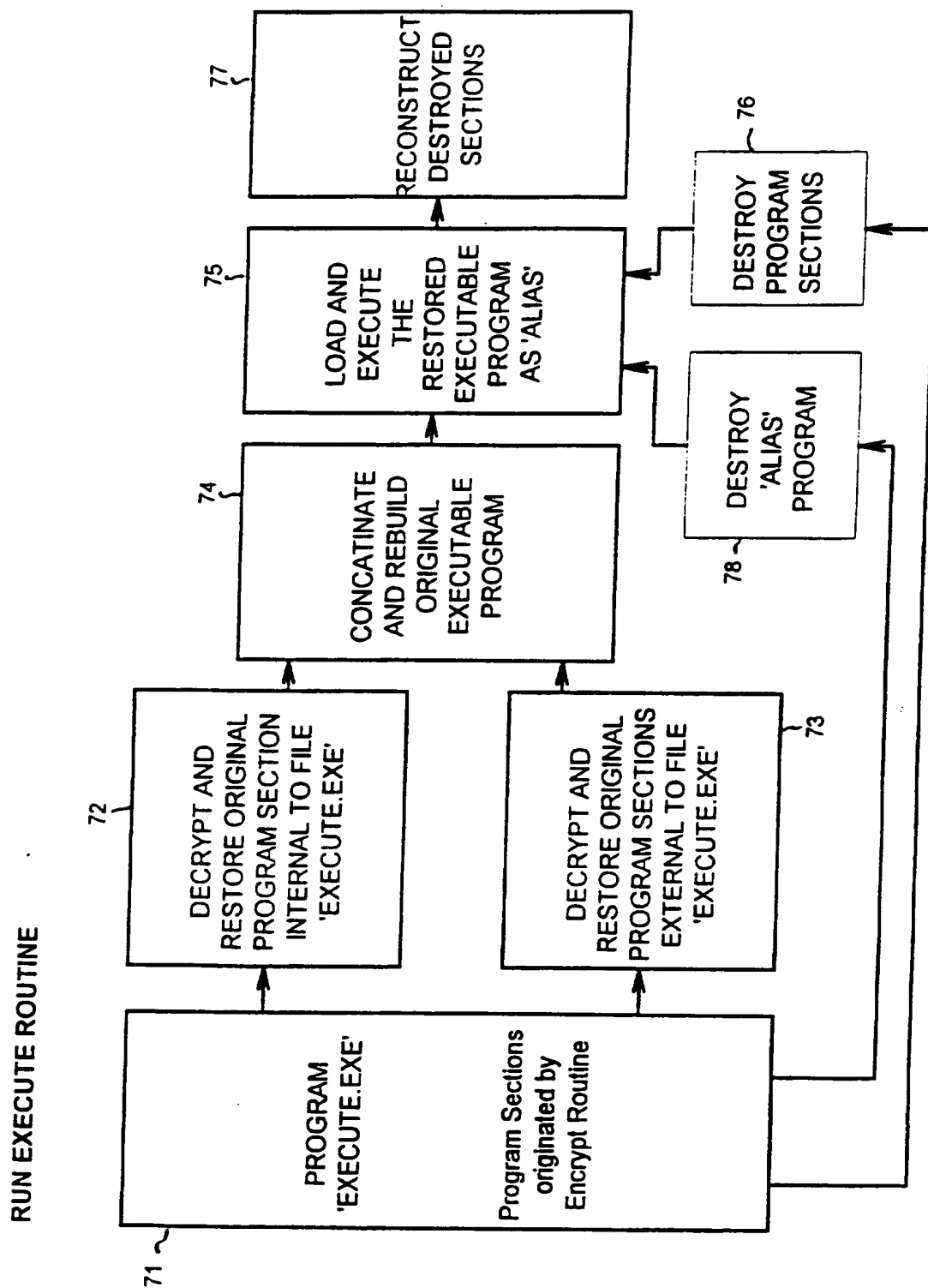
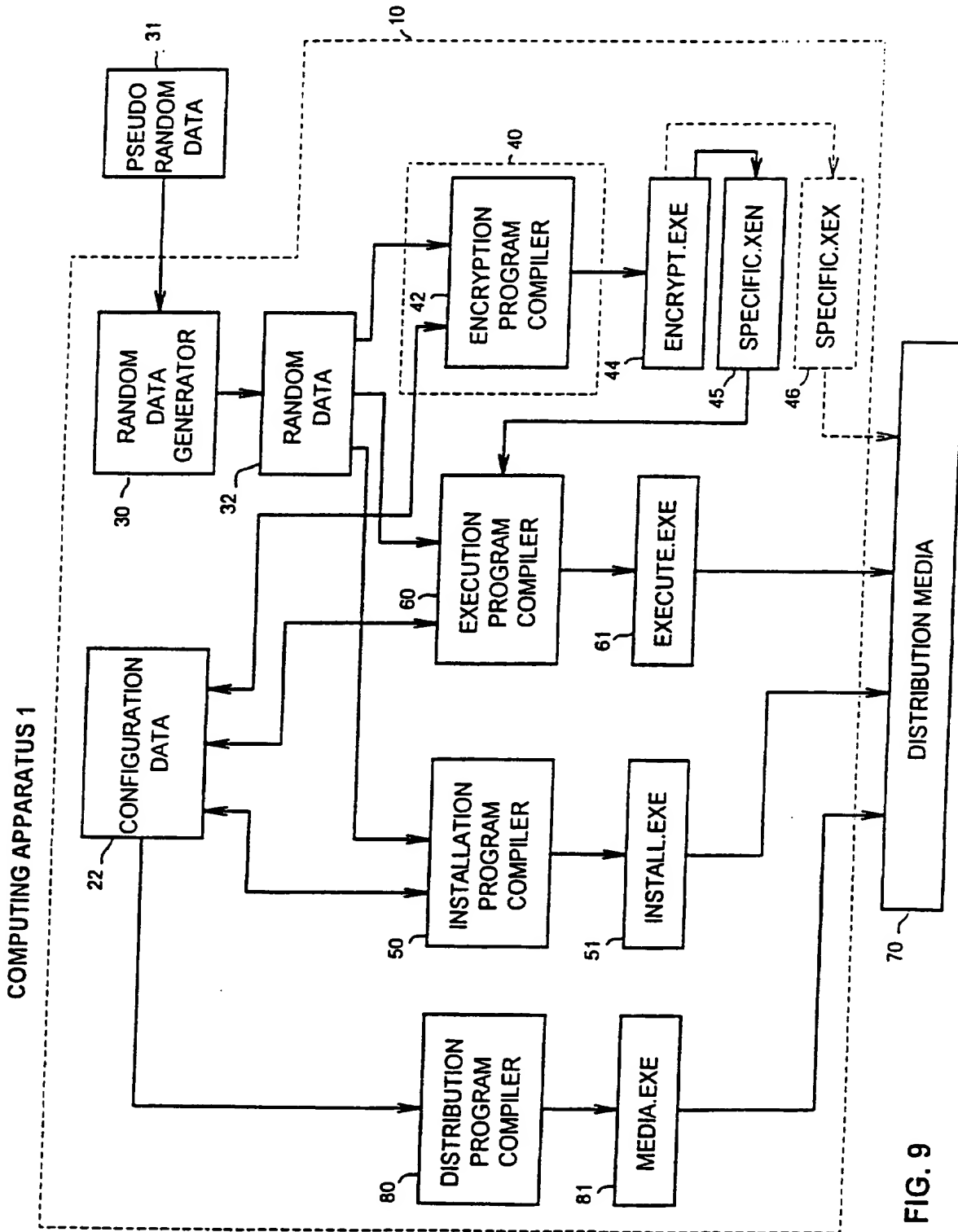
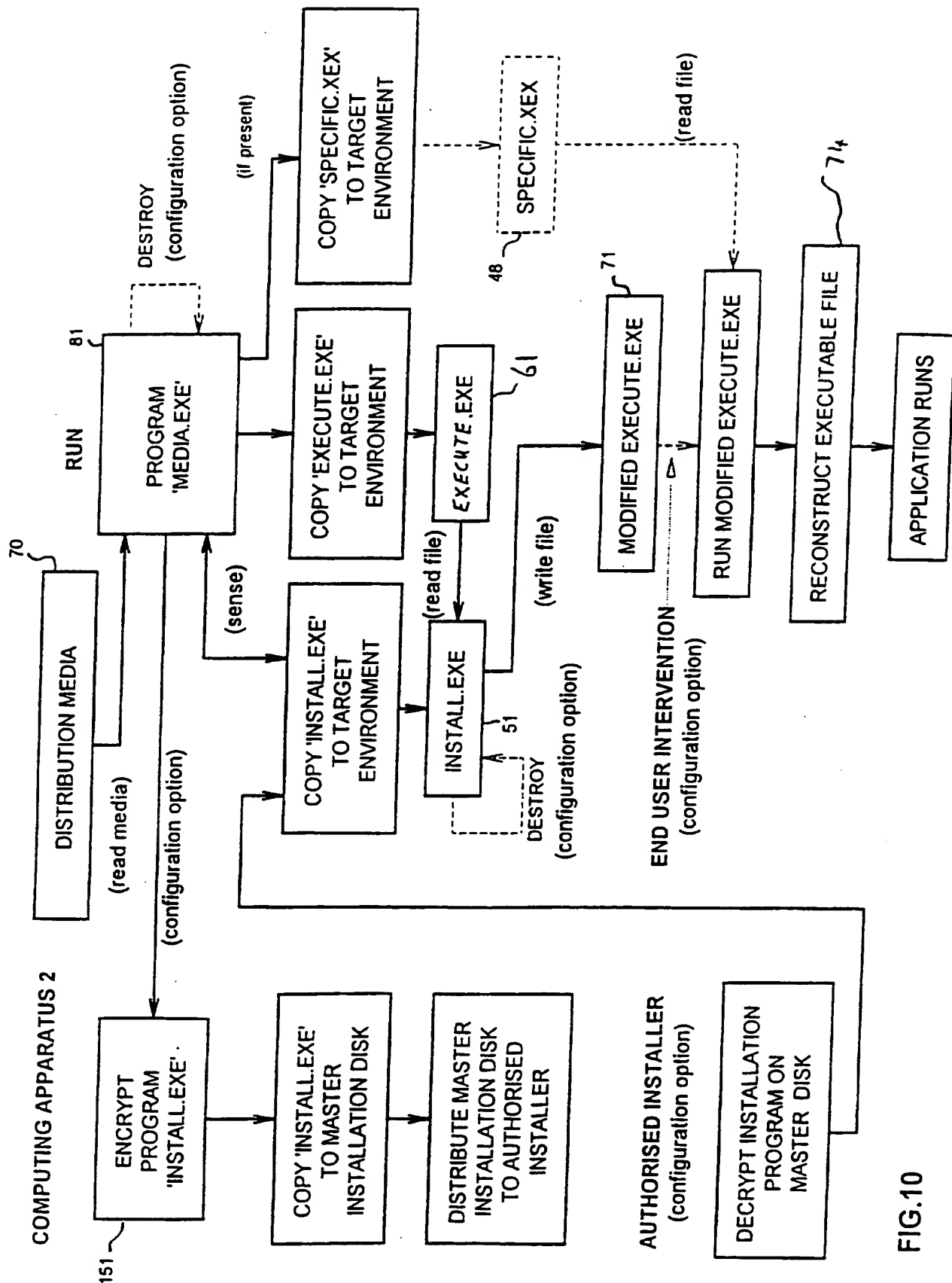


FIG. 8



10 / 10



INTERNATIONAL SEARCH REPORT

International Application No.

PCT/AU 95/00836

A. CLASSIFICATION OF SUBJECT MATTER

Int Cl^B: G06F 12/14, 19/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC : G06F 12/14, 19/00

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

US class : 380/4

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	AU 14856/95 A (IBM CORPORATION) 2 November 1995 See the whole document	1-36
P,A	EP 679978 A1 (IBM CORPORATION) 2 November 1995 See the whole document	1-36
A	US 5343527 A (MOORE) 30 August 1994 See the whole document	1-36



Further documents are listed in the continuation of Box C



See patent family annex

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance
 "E" earlier document but published on or after the international filing date
 "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
 "O" document referring to an oral disclosure, use, exhibition or other means
 "P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
 "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
 "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
 "&" document member of the same patent family

Date of the actual completion of the international search

14 March 1996

Date of mailing of the international search report

27 MARCH 1996

Name and mailing address of the ISA/AU
 AUSTRALIAN INDUSTRIAL PROPERTY ORGANISATION
 PO BOX 200
 WODEN ACT 2606
 AUSTRALIA Facsimile No.: (06) 285 3929

Authorized officer

R.W.J. FINZI

Telephone No.: (06) 283 2213

INTERNATIONAL SEARCH REPORT

International Application No.

PCT/AU 95/00836

C (Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5182770 A (MEDVECZKY et al) 26 January 1993 See the whole document	1-36
A	US 4999806 A (CHERNOW et al) 12 March 1991 See the whole document	1-36
A	GB 2146149 A (SOFTWARE DISTRIBUTION NETWORK INC.) 11 April 1985 See the whole document	1-36

Information on patent family members

PCT/AU 95/00836

This Annex lists the known "A" publication level patent-family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

Patent Document Cited in Search Report				Patent Family Member			
AU	14856/95	BR JP	9501522 7295803	CA	2145925	EP	679979
EP	679978	CA	2143874	JP	7295799		
GB	2146149	US	4550350				

END OF ANNEX

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

THIS PAGE BLANK (USPTO)